

дослідженні складних нелінійних моделей, що використовуються при дослідженні складних об'єктів до яких відносяться харчові продукти.

Список літератури: 1.Методы исследований и организация экспериментов [Текст] / под ред. проф. К.П. Власова. – Х.: Гуманитарный центр, 2002. – 256 с. 2.Дрейпер, Н. Прикладной регрессионный анализ [Текст] / Н. Дрейпер, Г. Смит – М.: Вильямс, 2007. – 912 с. 3.Хартман, К. Планирование эксперимента в исследовании технологических процессов [Текст] /. К. Хартман и др. – М.: «Мир». 1977, 552 с. 4.Льяконов, В.П. Mathcad 11/12/13 в математике [Текст] / В.П. Льяконов // Справочник. – М.:Горячая линия – Телеком, 2007. – 928 с. 5.Гантмахер, Ф.Р. Теория матриц [Текст] / Ф.Р. Гантмахер М.: «Наука». 1967, 576 с.

Поступила в редколлегию 12.05.2012

УДК 65.011.56

И. А. КРИВОРОТЕНКО, студ., ХНУРЭ, Харьков,
Е. П. ПАВЛЕНКО, канд.техн.наук, доц., ХНУРЭ, Харьков

РАЗРАБОТКА WEB-ОРИЕНТИРОВАННОГО ПРИЛОЖЕНИЯ УПРАВЛЕНИЯ ПРОЕКТАМИ КОМПАНИИ «PEOPLESOFT»

Розглянута проблема обліку розробки програмного забезпечення компанії «PeopleSoft». Досліджені основні тенденції розробки Web-додатку з використанням мови високого рівню Java, також проведено порівняння фреймворку Spring та технології Enterprise Java Beans.

Ключові слова: управління проектами, Spring framework

Рассмотрена проблема учета разработки программного обеспечения компании «PeopleSoft». Исследованы основные тенденции разработки Web-приложений с использованием языка высокого уровня Java, а также проведено сравнение фреймворка Spring и технологии Enterprise Java Beans.

Ключевые слова: управление проектами, Spring framework

It was considered the problem of accounting for software development company «PeopleSoft». It was researched the main trends of development Web-oriented applications with using high-level language Java, also it was compared Spring Framework and Enterprise Java Beans Technology.

Keywords: project management, Spring Framework

1. Введение

В настоящее время с целью эффективного управления бизнесом предприятия все шире используют информационные технологии. Внедрение информационных технологий в работу предприятия позволяет автоматизировать ряд бизнес-процессов, сократить время на их выполнение, а также повысить качество производимой продукции или предоставляемых услуг.

В последнее время очень популярными являются веб-ориентированные системы, которые позволяют эффективно и своевременно обеспечивать сотрудников необходимой информацией. Веб-ориентированные системы подразумевают, что вся бизнес-логика вынесена на сервер, а клиенту предоставляется только интерфейс. Для реализации этой части приложения можно использовать язык Java, который хорошо зарекомендовал себя в разработке корпоративных веб-приложений.

2. Постановка задачи

Компания «PeopleSoft» занимается разработкой программного обеспечения для различных сфер деятельности. Компания работает с большим количеством программных проектов, и в связи с этим возникает необходимость в автоматизации процесса их учета. Разработанное приложение позволит вести

учет новых проектов, а также получать оперативную информацию о состоянии существующих проектов.

Входной информацией разрабатываемого приложения являются: информация о сотрудниках, информация о задачах и информация о проектах.

Выходная информация: форма со списком всех сотрудников, а также аналогичные формы со списками задач и проектов, форма со списком задач для конкретного проекта и форма со списком задач для сотрудника в конкретном проекте. Все атрибуты выходных сообщений для сотрудников, задач и проектов соответствуют атрибутам входных сообщений.

Использование веб-технологий для разработки такого приложения дает ряд преимуществ, таких как:

- распределенный доступ для всех пользователей;
- доступ к информации через Интернет;
- установка приложения выполняется только на сервере;
- приложение не требует затрат на дополнительное ПО;
- приложение может работать на различных операционных системах;
- минимальные требования к мощности ПК.

Учитывая вышеперечисленные преимущества, было принято решение о разработке веб-ориентированного приложения управления проектами. Для разработки приложения была выбрана платформа JavaEE языка программирования Java. JavaEE позволяет строить эффективные серверные приложения, в которых необходимы гибкость, масштабируемость и надежность. Согласно рейтингу языков программирования, который составила голландская компания TIOBE Software BV, Java занимает первое место с 2002 года. На момент составления рейтинга 2011 года доля рынка данного языка программирования составила 17,4%.

3. Основные тенденции разработки Web-приложений с использованием Java

Изначально Web-ориентированные приложения, разрабатываемые на языке Java, строились с помощью сервлетов и технологии JSP. Сервлет – это java-класс, который выполняется на стороне сервера. Сервлет получает какую-либо информацию от клиента посредством http-запросов, обрабатывает на сервере и отправляет клиенту обратно посредством http-ответа. JSP – это технология, позволяющая встраивать java-код в гипертекстовые страницы. Страницы JSP состоят из двух составляющих: статического содержимого в виде html или xml и динамического – в виде JSP-компонентов.

На сегодняшний день данный подход считается устаревшим и неэффективным. Можно выделить два основных направления разработки Web-приложений:

- с использованием Enterprise Java Beans 3.0 (EJB3);
- с использованием Spring 3.0.

EJB – это управляемая технология на основе компонентной серверной архитектуры, предназначенная для написания и поддержки серверных компонентов, содержащих бизнес-логику. Эта технология является основным

элементом платформы JavaEE, которая используется для серверного программирования на Java.

Версия EJB 3 претерпела существенные изменения по сравнению с версией 2. Прежде всего, это возможность использования аннотаций, которые позволяют разработчику использовать декларативный стиль программирования. Также спецификация EJB версии 2 накладывала на разработку существенные ограничения – существовали определенные правила описания: для создания компонентов java-beans необходимо было применять наследование от некоторого класса и переопределять необходимые методы. В версии 3 многие из этих ограничений отсутствуют по причине введения аннотаций, которые позволяют декларативно описывать компоненты java-beans.

Альтернативным решением EJB является фреймворк Spring. Spring - это аспектно-ориентированный контейнер и фреймворк. Spring 3.0 также отличается от предыдущей версии использованием аннотаций, которые позволяют значительно уменьшить количество кода. До версии 3.0 компоненты Spring описывались при помощи языка разметки xml, что принуждало разработчика к написанию большого объема кода для описания и связывания компонентов.

Оба этих фреймворка имеют много общего: и тот, и другой используют POJO-классы для реализации бизнес-логики приложения. POJO-классы – это простые java-классы, не унаследованные от какого-либо интерфейса. Фреймворки связывают различные POJO-классы между собой при помощи внедрения зависимости. Таким образом, разработчик может уделить больше внимания реализации бизнес-логики конкретного приложения и unit-тестированию POJO-классов без использования фреймворков.

Внедрение зависимости – это реализация принципа объектно-ориентированного программирования «Обращения контроля (Inversion of control)», суть которого заключается в том, что все классы должны быть как можно более изолированными от других, не полагаясь в своей работе на детали конкретной реализации других классов. Данное понятие подразумевает связывание одного компонента с другим путем получения второго через конструктор, setter-метод или интерфейс внедрения. Класс получает на вход метода или конструктора не конкретный класс, а некоторый интерфейс, который могут реализовывать множество классов. Такое связывание между классами ещё называют слабым связыванием.

Хотя эти фреймворки очень похожи, они все же имеют некоторые различия. Основные различия состоят в подходе при разработке. При использовании Spring все классы, как для реализации сущностей, так и для бизнес-логики имеют вид POJO-класса. В EJB 3.0 все классы делятся на три вида: объектные (Entity Bean), сессионные (Session Bean), а также управляемые сообщениями (Message Driven Beans). Эти классы имеют различное назначение, что заставляет разработчика строить приложения, подстраиваясь под эти требования, в отличие от Spring, где все классы являются пользовательскими POJO-классами и связаны между собой. Ещё одна особенность Spring – использование аспектно-ориентированного программирования (АОП).

АОП – это новый подход к программированию, который подразумевает выделение общей функциональности в отдельный модуль. Под общей функциональностью понимаются некоторые компоненты, которые используются в различных модулях системы. АОП позволяет декларативно определить, как и в каких модулях системы применять эту функциональность без внесения изменений в модули. Как пример такой функциональности можно привести ведение логов или управление транзакциями. Таким образом, АОП дополняет объектно-ориентированное программирование, обогащая его другим типом модульности, который позволяет локализовать код сквозной логики в одном модуле.

Пример использования АОП и других преимуществ Spring представлен ниже:

```
package my.project.dao;
import java.util.Collection;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;
import org.springframework.transaction.annotation.Transactional;
import my.project.domain.Entity;
/**
 * @Version 1.0 29.04.2012
 * @author Ivan Krivorotenko
 */
@Component
public class HibernateEntityDao implements EntityDao {
    @Autowired
    private SessionFactory sessionFactory;
    public SessionFactory getSessionFactory() {
        return sessionFactory;
    }
    public void setSessionFactory(SessionFactory sessionFactory) {
        this.sessionFactory = sessionFactory;
    }
    @Transactional
    public void create(Entity entity) {
        Session session = getSessionFactory().getCurrentSession();
        session.save(entity);
    }
    ...
}
```

В данном примере описывается класс, который описывает класс доступа к данным для некоторой сущности. Класс помечен аннотацией `@Component`, которая говорит о том, что он будет помещен в Spring контейнер. Поле `sessionFactory` помечено аннотацией `@Autowired`, которая позволяет

декларативно связывать классы друг с другом. Класс SessionFactory отвечает за создание и контроль над сессией.

В классе примера реализован метод create, который позволяет создавать некоторую сущность. Аннотация @Transactional заменяет открытие и закрытие транзакций в теле метода – это все будет выполнено автоматически. Для этой задачи Spring использует соответствующие аспекты, которые выполняются до и после метода. Остальные методы реализуются аналогично.

4. Выводы

EJB имеет репутацию фреймворка, который используется для построения крупных корпоративных Web-приложений. Также для развертывания EJB-приложений требуется серьезный сервер приложений вроде JBOSS, использование простого JSP/Servlet контейнера Apache Tomcat не поддерживается.

Приложение управления проектами является небольшим по своим размерам, и рассчитано на использование внутри компании. Поэтому можно сделать вывод, что разработка Web-приложения с использованием EJB нецелесообразна. Нет необходимости разворачивать сервер приложений, т. к. можно обойтись простым контейнером Apache Tomcat. Также Spring предлагает свои библиотеки JSP тегов, которые позволяют строить пользовательский интерфейс без использования сторонних фреймворков.

Список литературы: 1. Walls, C. Spring in Action. Third Edition [Текст] / C. Walls – Manning Publications Co., 2011.- 426 с. 2. Panda, D. EJB 3 in Action [Текст] / D. Panda, R. Rahman, D. Lane – Manning Publications Co., 2007 – 677 с.

Поступила в редколлегию 21.05.2012

УДК 004.538

А.В. BOBK, канд.техн.наук, ст. преп., ХНУРЭ, Харьков,
Ю.А. ШЕВЧЕНКО, студ, ХНУРЭ, Харьков

ВНЕДРЕНИЕ КРОСС-МЕДИЙНЫХ ТЕХНОЛОГИЙ НА ПОЛИГРАФИЧЕСКИХ ПРЕДПРИЯТИЯХ

У даній роботі розглядається поняття крос-медійний видавничий процес, події попередні його виникнення, його переваги на ринку та шляхи розвитку в подальшому. Наведено схема підприємства до і після впровадження крос-медійних технологій.

Ключові слова: крос-медійне видання, веб-сайт, цифровий друк.

В данной работе рассматривается понятие кросс-медийный издательский процесс, события предшествующие его возникновению, его преимущества на рынке и пути развития в дальнейшем. Приведена схема предприятия до и после внедрения кросс-медийных технологий.

Ключевые слова: кросс-медийное издание, веб-сайт, цифровая печать.

In this paper the concept of cross-media publishing process, the events preceding its emergence, its advantages in the market and the development in the future. A scheme of the company before and after the introduction of cross-media technologies.

Keywords: cross-media publishing, website, digital printing.